

# Getting Graphs a Good Look: Schemes and the Graph Editor

Bill Rising

StataCorp LP

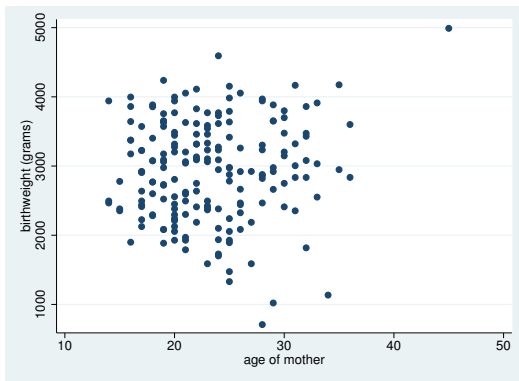
First Portuguese Stata Users Group Meeting  
Braga, Portugal  
September 17, 2010

# Situation

- Like Stata graphs
- Would like to make them look different
  - Needed for a particular journal
  - Needed for in-house graphics
  - Just because

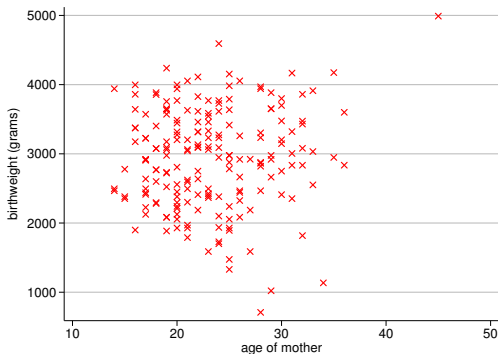
# The Default Look

- We have a default graph from Stata



# A Better Look

- We would like it to look like this, instead

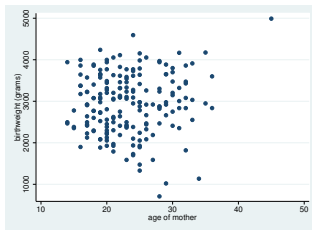


# Three Solutions

- The brute force solution
  - Have a series of options we give to every Stata graph
  - This is not a good way to work!
- Use the Graph Recorder to change one graph, then apply the changes to other graphs
  - Simple via point and click, but needs many versions
- Write a *scheme* to control the look of all Stata graphs
  - A little more complex, but works more generally
- We'll now go through each of these

# Our Original Graph

- We'll use the `lbw` dataset from Stata's manuals
  - . `webuse lbw`
- Start with the simple birthweight vs. age
  - . `scatter bwt age`



- We'll change the background, the plot symbols, the grid lines, and the y-axis labels

## Recordings for Simple Changes

- Recordings can be used to make common changes, especially if there is a single type of graph you make often
- Be forewarned, however, that a recording for a twoway graph will most likely not work for, say, a histogram
  - This means one recording would be needed for each type of graph...

## The Record Button

- Start up the Graph Editor
- Click the red **Start Recording** button to record everything done
- We'll change just overall characteristics here
  - We could record added text, lines or points
  - These would be positioned by the units in the graph, though



## Change the background to white

- Double click outside the graph
- In the *Graph properties* dialog box, change the color to *white*
- Click **OK**

## Changing Plot Points to X's

- Right-click on one of the points
- From the contextual menu choose *Scatter properties*
- Change the symbol to *Large X*
- Click the **OK** button

## Darkening the Gridlines

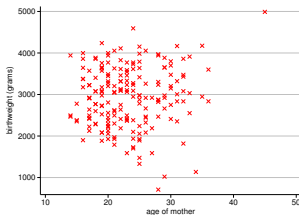
- Double-click just to the left of the vertical axis
- Click the **Grid lines** button
- Choose the *Gray 12* color
- Click the **OK** button

## Making the $Y$ -axis Ticks Horizontal

- Click the **Label properties** button
- Select the *Horizontal* item from the *Angle* field
- Click the **OK** button
- Click the **OK** button in the *Axis properties* dialog box

# Save the Recording

- Click the Graph Record button again, to save the recording
- Give it some nice name, like `x grid`
- Close the graph editor but do not save the recording



## Playing the Recording in the Graph Editor

- Reissue the `twoway` command from earlier
- Start the Graph Editor
- Click the **Play** button
- Select the `x grid` item
  - The graph recording gets played
- Close the Graph Editor

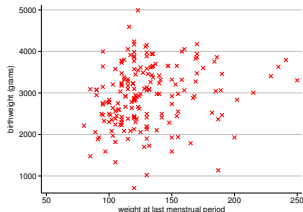
## Playing the Recording from the Command Window

- Get your twoway command
- Add an `play(x grid)` option
- Issue the command
  - The graph recording is applied to the graph

# Playing with Other Data

- This can be applied to other scatterplots well enough

```
. scatter bwt lwt, play(x grid)
```



- So...we have a way of changing simple scatterplots



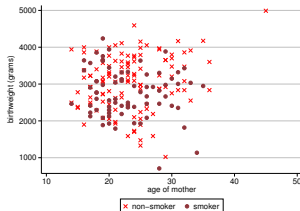
## Comments on Graph Recordings

- Your graph recordings are saved in your **PERSONAL** folder
- They can be quite useful for such schemelets
- If you look at them, you will see that they work directly with the graphics class system—edit at your own risk

## More Complex Graphs

- If we have overlaid graphs, the changes apply to just the first plot

```
. twoway (scatter bwt age if smoke==0) ///  
.   (scatter bwt age if smoke==1), ///  
.   legend(order(1 "non-smoker" 2 "smoker")) ///  
.   play(x grid)
```



# What to Do?

- So, if we would like something which applies to deeper graphs we have two choices
  - Make another recording for the more complex graph
  - Try to make more general changes using a scheme
- We'll now look at schemes

# What is a Scheme?

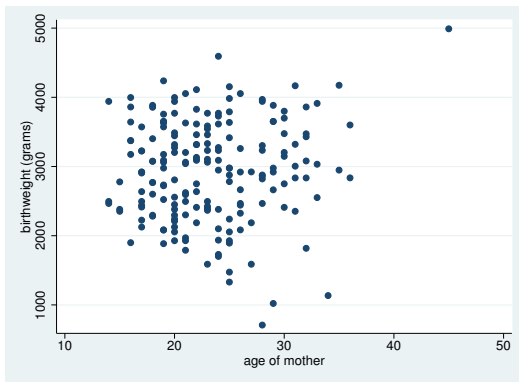
- A scheme is a series of settings which tell Stata how to draw graphs
- To see what schemes you have, try this
  - What is listed depends on what you might have installed
  - I have some extras here for showing off
- We should try a few

# Why Use a Scheme?

- There is a standard appearance needed
  - For in-house graphics
  - For favorite journals
  - For your field
- You would like a different look
- You would like to implement this look without having to specify a long series of options to every graph command you use

## Our Model Graph

- We'll keep working with our simple graph from before, as a start
  - . scatter bwt age



# Trying Some Schemes

- By default, the `s2color` scheme is used in Stata
  - This could be change in the graph preferences
- Here is the black-and-white version:  

```
. scatter bwt age, scheme(s2mono)
```
- Or...reverse video  

```
. scatter bwt age, scheme(s1rcolor)
```
- Or a very plain yet practical scheme, which is user-written  

```
. scatter bwt age, scheme(lean2)
```
- Here is something ugly, stolen from a colleague at work  

```
. scatter bwt age, scheme(ugly)
```

## How are Schemes Implemented?

- Schemes are simple text files which list attributes of graphs, and then list how they should appear
- We'll base our scheme on `s2color` and just change what we like
  - This is the best way to work
- We'll then save the scheme with a very special name  
`scheme-scheme-name.scheme`
- Stata will find the scheme if it is anywhere in the ado-path



## What Do We Have?

- Redraw the scatterplot from above using the default `s2color` scheme
  - . `scatter bwt age`

# What would we like to change?

- We'll start by changing what we changed in the editor
  - A white background
  - X's for the plot symbols
  - Darker grid lines
  - Horizontal labels on the vertical axis

# Strategy

- We'll work with on-line help, because
  - Schemes are ...
    - ... simple: each line has an attribute, a context and its value
    - ... complex: the values can be raw or part of a style
- We'll pick out our changes as we need them from the online help:  
help scheme entries
  - This is a rare example of help which is on-line but *not* in the manuals
- We'll use the Command window only
  - Reloading new versions of a scheme clears out the dialog boxes

# Starting the Changes

- Open a new Do-file Editor window
- We'd like to alter the `s2color` scheme, so start by including this in the scheme:

```
#include s2color
```

Be sure there is no space after the hash (#) mark!

- This will base our scheme on Stata's `s2color` scheme
- Save the scheme as `scheme-xgrid.scheme` in your scheme folder

## Finding What to Change

- If you didn't do it earlier, type `help scheme entries` to get to the place which says how to change scheme entries
- We'd like to change the look of our scatterplots, so click the `Scatter plots` link
  - This looks understandable, except the for `p#` entries

# What Do $p$ and $p\#$ Mean?

- Because Stata allows overlays of graphs, it needs methods for rendering each of the graphs
  - There are 15 sets of settings—after the fifteenth overlay, the settings recycle
- Use  $p$  if the change pertains to all overlays of this type
- Use the  $p\#$  if the change pertains to a particular overlay

## Making Plot Symbols into Xs: Help

- We'd like all the plot symbols to be large X's
  - It looks like `symbol1` is what we're after
  - We can use `p` to universally change to X's
- Clicking the `symbolstyle` link gives lists of things to try
  - We can try `x` for large X
- Add this line to your scheme file:

```
symbol p x
```

## Changing the Symbol Color

- There is (apparently) no entry for changing the color of the plot symbols
- When this happens, take a look at `help scheme shared plots`
- It looks like we need to add

```
color p1 red
```

to make the plot symbol red for the first plot

- Save your scheme file



## Trying Out Our Creation

- Try from command line:  
`. scatter bwt age, scheme(xgrid)`
- The symbols are now red X's

## Changing the Gridlines

- We'd like the gridlines to be more visible
- Bring the Viewer window to the front
- Click the **Back** button to get back to the help scheme entries page
- Click the gridlines link
  - It looks like `color major_grid` is what we'd like to change
  - Click the `colorstyle` link to see how colors work

# Colors in Stata: Names and Numbers

- There are several alternatives
  - Named colors
  - 8-bit RGB colors
    - Colors are specified as *Red# Green# Blue#*; each is a number from 0 to 255
  - 8-bit CMYK colors
    - Colors are specified as *Cyan# Magenta# Yellow# Black#*, with each number from 0 to 255

## Changing the Grid Color

- Let's try changing the color of the grid line to a named color
- Add this line to your scheme

```
color major_grid gs12
```

- Save the scheme file

## Trying Out Our Second Creation

- Try reissuing the our command from earlier. . .you'll find nothing has changed

```
. scatter bwt age, scheme(xgrid)
```

  - This is because the scheme has been cached by Stata
- Flush the scheme from Stata's cache

```
. discard
```
- Reissue the graph command, again

```
. scatter bwt age, scheme(xgrid)
```
- Stata will re-read the scheme file, so the new changes will be incorporated

## Help for Changing the Y-axis Labels

- We'd now like to make the axis labels horizontal
- Bring the Viewer window to the front
- Click the **Back** button to get back to the help scheme entries page
- Click the Axes link
- Scroll to the “Axis tick labels” section
- It looks like we need to change the text angle for the y axis
- Click the link to `anglestyle` link
- We need to change the angle style to horizontal

## Changing the Y-axis Labels

- Add the following line to your scheme  
`anglestyle vertical_tick horizontal`
- Save the scheme file
- discard the cache
- Reissue our favorite command  
`. scatter bwt age, scheme(xgrid)`
- The y-axis labels are now horizontal

## Changing the Background

- To find this, go back to `help scheme` entries and scroll until you find the `Background` style definitions
- We just need to add  

```
color background white
```

to the scheme file
- Discard and try the graph again



## We Have Our Scheme

- This is now a `xgrid` scheme we could use when needed
- We could add to it as we see fit to make a full set of changes
- The strategy we used is pretty much what everyone needs
- If we wanted to make this our default scheme, we could use the `set scheme` command:

```
set scheme xgrid [ , permanently ]
```

## Extending the Scheme to Histograms

- Now suppose that we would like to play with histograms
  - Specifically, it would be nice to change the default color of the bars
- Hit the **Back** button in the Viewer and click the Histogram plot link
- We'd like to change the color of the bars to a nice blue:
  - `color histogram "40 40 255"`
- If you try this graph, you'll see the outlines of the bars are now ugly
- Add one more line
  - `color histogram_line black`
- Now the histograms look nice, also

# A Complete Scheme

- To find a full scheme, use Stata's `viewsource` command:  
`viewsource scheme-s2color.scheme`
- There is a lot here!

## A Carefully Built Scheme

- To find how a nice scheme can be made without starting from scratch, look at one of the *lean2* schemes from Svend Juul

```
findit lean2 scheme
```

- Click the first link
  - Install the files
- Use `viewsource` to look at the file

## Where to Save?

- We saved this scheme in the folder we're using
- This makes it usable only when working in this folder
- To use it other places, it needs to go in a location in Stata's *ado-path*

# Conclusion

- There are two ways to uniformly alter the look of Stata graphs
- Graph recordings
  - Are quick to make
  - Change specific graph types, so many would be needed
- Schemes
  - Take more time to make
  - Change all graph types